

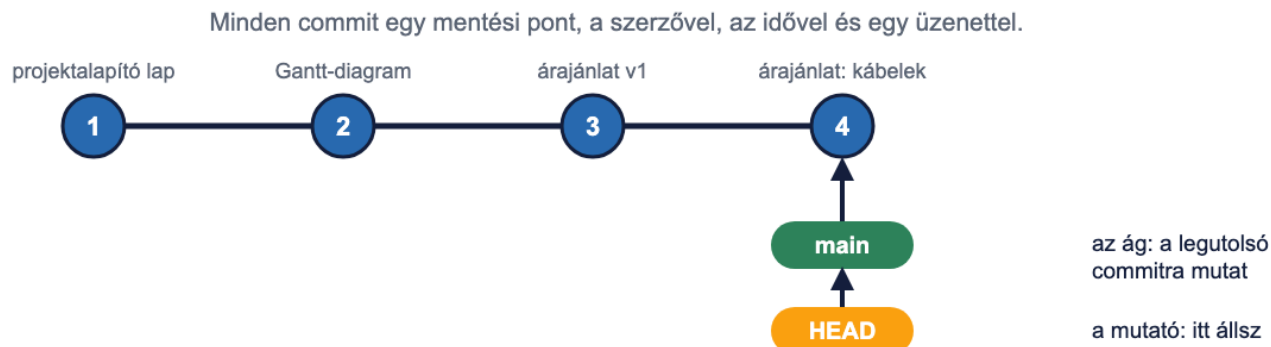
GitHub: fogalmak és tipikus hibák

Segédlet a csapatmunkához. Ha valami nem úgy megy, mint vártad, itt keresd meg!

1. rész – A legfontosabb fogalmak

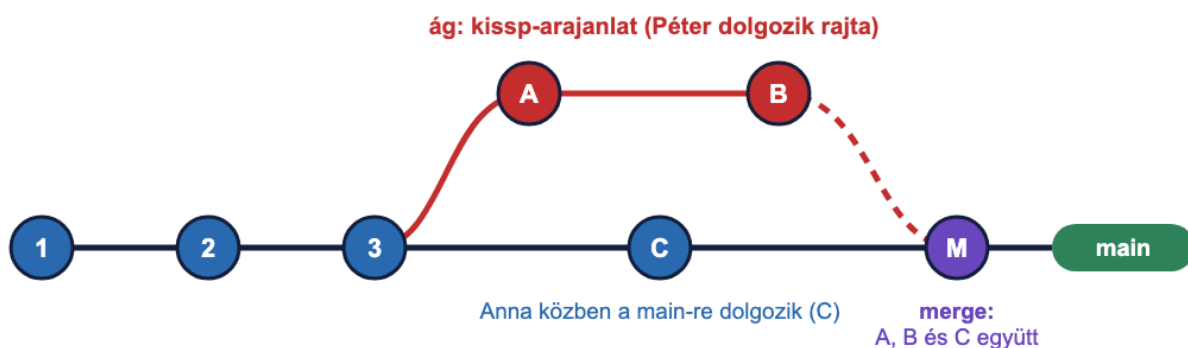
Fogalom	Mit jelent?	Hasonlat
Tároló (repository, repo)	a projekt mappája a GitHubon, az összes fájljal és a teljes történetükkel	a csapat közös szekrénye
Commit	egy mentési pont: „ezt és ezt változtattam”. Mindegyiknek van szerzője, ideje és üzenete.	egy bejegyzés a csapat naplójában
Commit-üzenet	a változás rövid leírása, pl. „ <i>árajánlat: kábelek ára hozzáadva</i> ”	a naplóbejegyzés szövege
Ág (branch)	a munka egy külön szála. A fő ág neve main .	egy piszkozat, amin kísérletezhetsz anélkül, hogy a „tisztázatot” elrontanád
Mutató (HEAD)	megmutatja, melyik commitnál „állsz” éppen . Az ág maga is egy mutató: mindig az utolsó commitjára mutat.	a könyvjelző a naplóban
Merge (összefésülés)	két ág változásainak egyesítése	két piszkozatból egy tisztázat lesz
Pull request (PR)	kérés: „nézzétek át a változásaimat, és fésüljük be a main-be”	beadod átnézésre a munkádat
Clone	a tároló letöltése a saját gépre	lemásolod a szekrény tartalmát
Push	a saját gépen készült commitok feltöltése a GitHubra	visszateszed a szekrénybe
Pull	a GitHubon lévő újdonságok letöltése a saját gépre	megnézed, mit tettek be a többiek
Fork	egy más által készített tároló lemásolása a saját fiókodba	
Collaborator (közreműködő)	akit meghívtak a tárolóba, és írhat bele	akinek van kulcsa a szekrényhez

Hogyan néz ki a mutató és az ág?



Ha új commitot készítesz, a **main** és a **HEAD** is egy lépéssel előrelép.

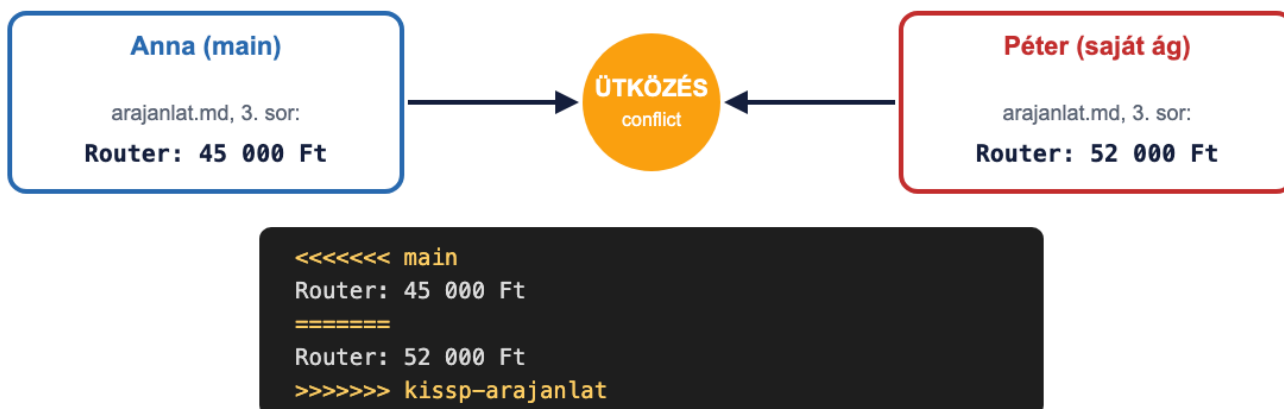
Két ág, majd összefésülés



2. rész – Tipikus hibák és a megoldásuk

Merge conflict (ütközés)

Ketten ugyanannak a fájlnak ugyanazt a sorát írták át. A Git nem tudja, melyik a jó.



Mi történt? Két ember **ugyanannak a fájlnak ugyanazt a sorát** változtatta meg, és a Git nem tudja eldönteni, melyik a jó. **Így néz ki a fájlban:**

```
<<<<<<< main
Router ára: 45 000 Ft
=====
Router ára: 52 000 Ft
>>>>>>> kissp-arájanlat
```

- A <<<<<<< és a ===== között az egyik változat van, a ===== és a >>>>>>> között a másik.

Megoldás:

1. Beszéljétek meg, melyik a jó (vagy mindkettőből kell valami).
2. Írd át a részt úgy, ahogy végül lennie kell, és **töröld a három jelölősort** (<<<<<<<, =====, >>>>>>>).
3. Mentsd el, és commitold: „*ütközés feloldva: router ára 52 000 Ft*”.

A GitHub weboldalán egy pull requestnél a **Resolve conflicts** gomb ugyanezt mutatja meg a böngészőben.

Megelőzés: egy fájlban egyszerre egy ember dolgozzon. A felelőslistán legyen leírva, ki melyik fájlért felel.

Word, Excel és képfájlok: „nem látom a különbséget”

A Git a **szöveges** fájlokban (.md, .txt, .html, .py) soronként látja a változást. A .docx, .xlsx és .ppt fájlokba nem lát bele: ha ketten módosítják, **nem tudja összefésülni**, és az egyik változat felülírja a másikat. **Megoldás:** ezeken a fájlokon **egyszerre csak egy ember** dolgozzon. Előtte szóljon a csapatnak: „most én szerkesztem az árajánlatot”.

„Felülírtam a társam munkáját” vagy „rossz fájlt töltöttem fel”

Nem vészett el! Minden korábbi változat megvan.

- A fájlra kattintva → **History** → keresd meg a jó változatot → nyisd meg → **Download** (vagy a „...” menüben a nyers fájl), és töltsd fel újra.
- Commit-üzenet: „*visszaállítva a nov. 18-i változat*”.

„Töröltem egy fájlt”

Ugyanígy: a tároló **commits** listájában keresd meg az utolsó commitot, amelyben még megvolt, nyisd meg a fájlt, és töltsd fel újra.

Nem tudok feltölteni: nincs „Add file” gomb, vagy „Permission denied” / 403

Ok: nem vagy közreműködő a tárolóban, vagy még nem fogadtad el a meghívást. **Megoldás:** nézd meg az e-mailjeidet (a meghívó levél), vagy kérd meg a dokumentátort, hogy hívjon meg újra: Settings → Collaborators → Add people.

„File is too large” (túl nagy fájl)

A böngészőből egyszerre legfeljebb kb. **25 MB**-os fájl tölthető fel, a GitHub pedig a **100 MB**-nál nagyobb fájlokat egyáltalán nem fogadja el. **Megoldás:** tömörítsd a képeket, a videót tedd máshova (pl. OneDrive), és a tárolóba csak a linkjét írd.

Commit után nem látom a változást

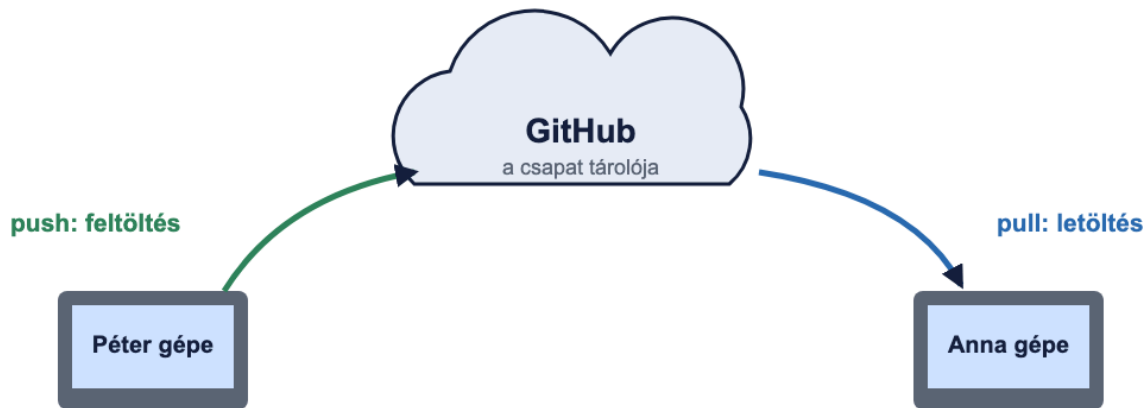
- Frissítsd az oldalt (**F5**).
- Biztosan a **Commit changes** gombra kattintottál, és nem csak bezártad az ablakot?

- Jó ágat nézel? A fájllista fölött bal oldalt látszik az ág neve: legyen **main**.

„Egy jelszót vagy személyes adatot töltöttem fel”

Azonnal szólj a tanárnak! A törlés nem elég, mert a történetben megmarad. A jelszót **meg kell változtatni**. Ezért szabály: jelszót, telefonszámot és lakcímet **soha** nem teszünk a tárolóba.

Parancssorból (ha később a Git programot használod a gépen)



clone = az első letöltés · a böngészős feltöltésnél ezek nem kellenek

Hibaüzenet	Mit jelent?	Megoldás
! [rejected] main -> main (fetch first)	a GitHubon van olyan újdonság, ami nálad még nincs	előbb git pull, aztán git push
CONFLICT (content): Merge conflict in arajanlat.md	ütközés (lásd fent)	oldd fel a fájlban, aztán git add arajanlat.md és git commit
nothing to commit, working tree clean	nincs új változás, amit menteni lehetne	ez nem hiba
You are in 'detached HEAD' state	a mutató (HEAD) egy régi commitra mutat, nem egy ágra	git switch main (vissza a fő ágra)
Authentication failed	a GitHub a parancssorban nem a jelszavadat kéri	használd a GitHub Desktop programot, vagy kérj segítséget a tanártól (token kell)

3. rész – A csapat 6 arany szabálya

1. **Kezdés előtt:** frissítsd az oldalt, és nézd meg, mit töltöttek fel a többiek.
2. **Egy fájl, egy ember:** szólj, mielőtt egy közös fájlt szerkesztesz.
3. **Kis lépésekben:** inkább sok kis commit, mint egy óriási a határidő előtt.
4. **Érthető üzenet:** „*árajánlat: kábelezés hozzáadva*”, és nem „*asdf*” vagy „*javítás*”.
5. **Semmi titkos adat:** jelszó, telefonszám, lakcím soha.
6. **Ha elrontottad:** ne ess pánikba. A történetből minden visszahozható. Szólj a csapatnak.